

## CLAIMS

We claim:

1. A method comprising:

receiving a playlist referencing a first media segment and a second media segment, the second media segment operable to play automatically without a prompt after being loaded;

presenting the first media segment; and

prerolling the second media segment.

2. A method as recited in claim 1 wherein prerolling the second media segment comprises:

loading at least a predetermined minimum portion of the second media segment before the presenting of the first media segment is complete; and  
postponing presenting of the second media segment.

3. A method as recited in claim 2 wherein the postponing playback comprises:

issuing a stop command to a control.

4. A method as recited in claim 2 wherein the postponing playback comprises:

stopping a timer associated with presenting the second media segment.

5. A method as recited in claim 2 further comprising:

1 receiving an event indicating that the presenting of the first media segment is  
2 finished; and  
3 in response to receiving the event, starting presentation of the second media  
4 segment.

5  
6 6. A method as recited in claim 2 further comprising:  
7 receiving a loading complete event indicating that the second media segment has  
8 been completely loaded; and  
9 in response to receiving the loading complete event, prerolling a third media  
10 segment.

11  
12 7. A method as recited in claim 1 wherein the second media segment  
13 comprises an interactive media segment.

14  
15 8. A method as recited in claim 1 wherein the second media segment is  
16 further operable to issue a custom event.

17  
18 9. A method as recited in claim 8 wherein the custom event references a third  
19 media segment to be played in response to the custom event.

20  
21 10. A method as recited in claim 5 further comprising receiving an end of  
22 playback event from the second media segment.

1           11.     A method comprising:  
2           parsing a playlist having at least one reference to an interactive media segment  
3 operable to play continuously;  
4           playing the interactive media segment in an interface of a host application with a  
5 control operable to play the interactive media segment; and  
6           receiving a media segment event from the control indicating that the playing of  
7 the interactive media segment has finished.

8  
9           12.     A method as recited in claim 11 further comprising:  
10          stopping playback of the interactive media segment; and  
11          playing a subsequent media segment referenced by a subsequent reference in the  
12 playlist.

13  
14          13.     A method as recited in claim 11 wherein the playlist comprises an ASX  
15 file.

16  
17          14.     A method as recited in claim 11 further comprising:  
18          issuing to the host application a host-recognized event corresponding to the media  
19 segment event.

20  
21          15.     A method as recited in claim 14 wherein the media segment event  
22 comprises an EndOfPlayback event and the host-recognized event comprises a  
23 WMPEndOfPlayback event.

1           16.     A method as recited in claim 11 wherein the media segment event  
2 comprises a custom event.

3  
4           17.     A method as recited in claim 11 further comprising:  
5           playing a first media segment prior to the interactive media segment; and  
6           buffering the interactive media segment in memory prior to completion of the first  
7 media segment.

8  
9           18.     A method as recited in claim 17 further comprising:  
10          receiving a buffer progress indication from the control, the buffer progress  
11 indication indicating that a predetermined minimum portion of the interactive media  
12 segment has been buffered;  
13          issuing an EndOfBuffering event to the host application.

14  
15          19.     A method as recited in claim 18 wherein the predetermined minimum  
16 portion is 100% of the interactive media segment.

17  
18          20.     A method as recited in claim 18 wherein the predetermined minimum  
19 portion is less than 100% of the interactive media segment.

20  
21          21.     A method as recited in claim 18 further comprising:  
22          receiving a buffer complete indicator from the control indicating that 100% of the  
23 interactive media segment has been buffered;  
24          issuing an EndOfStreaming event to the host application.

1           22.     A computer-readable medium comprising computer-executable  
2 instructions to perform a method comprising:  
3           instantiating an events wrapper associated with a Flash ® media segment;  
4           initializing a control operable to playback the Flash ® media segment;  
5           hosting the control in a portion of a user interface; and  
6           receiving notification from the control when the Flash ® media segment has  
7 completed playback.

8  
9           23.     A computer-readable medium as recited in claim 22 wherein the method  
10 further comprises:

11           buffering the Flash ® media segment prior to completion of playback of a  
12 previous media segment;  
13           receiving an end of buffering event from the control when the Flash ® media  
14 segment has finished buffering; and  
15           playing the Flash ® media segment when the previous media segment completes  
16 playback.  
17

18  
19           24.     A computer-readable medium of claim 22 wherein the method further  
20 comprises:

21           playing a later media segment after receiving the notification that the Flash ®  
22 media segment has completed playback.  
23  
24  
25

- 1 25. A system comprising:  
2 a media control operable to begin playing a media segment automatically after  
3 buffering the media segment; and  
4 a host application operable to receive a reference to the media segment, initialize  
5 the media control with the media segment, and cause the media control to postpone  
6 playing of the media segment after the media segment is buffered.  
7
- 8 26. A system as recited in claim 25 further comprising:  
9 an events wrapper operable to receive an end of buffering notification from the  
10 media control and issue a corresponding end of buffering notification to the host  
11 application.  
12
- 13 27. A system as recited in claim 25 further comprising a playlist having one or  
14 more references to media segments to be played in sequential order.  
15
- 16 28. A system as recited in claim 27 wherein at least one of the referenced  
17 media segments comprises mixed media.  
18
- 19 29. A system as recited in claim 27 wherein the playlist comprises an event  
20 name associated with an event media segment to be played when a referenced media  
21 segment issues an event having the event name.  
22  
23  
24  
25

30. A system as recited in claim 25 wherein the playing of the media segment  
is postponed at least in part by stopping a timer that sends timer ticks to the media control  
for advancing playing of the media segment.

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25